

CONTROL DATA® 6000 SERIES SYSTEMS

Multi-Function Central Processor ■ Ten Peripheral Processors ■ Concurrent Parallel Operation ■ Multi-Processing ■ Total Compatibility ■ Remote Hook-Up





PRIME MOTIVATIONAL FORCES FOR THE INDUSTRY

The world's fastest, most versatile computing system, the **Control Data 6600**, is now in operation on-site, setting the pace for systems to follow.

It is a complex of computers and peripheral equipment with unprecedented operating modes and capabilities for multi-processing. It can handle 3,000,000 instructions per second, requiring some of its circuits to switch back and forth more than 40 million times a second. It separates peripheral control from central processor functions paving the way to a multitude of uses, from the very large, complex problem to the time-sharing of many smaller problems.

Compared with other systems that have only a fraction of this speed and capability, it is deceptively compact and simple in appearance, as well as being easy to use. Eleven computers are packed into four cabinets joined in the form of a cross. Ten are used for peripheral and executive control, leaving one for unhampered central processing.

Special software simplifies programming for all processors involved, and the operator need only contend with two cathode-ray tube displays and a 50-key typewriter keyboard for full control.

In all modesty, the **Control Data 6600** is truly an amazing hardware/software package, but in all honesty, what you have read so far is old news.

Approaching the Ultimate

The truth of the matter is that Control Data designers have found a way to extend the benefits of this new concept in logic and parallel operation. The result? Two additional systems: The **Control Data 6400** and **Control Data 6800**. All three sys-

tems are totally compatible and operate on the same basic principle, providing a prime motivational force that challenges the creative imagination of system designers and programmers to explore new areas of data processing and new solutions to old problems.

They need not depart from traditional methods of operation, nor wait any longer, to execute any number of highly sophisticated multi-processing, parallel operations based on three levels of use:

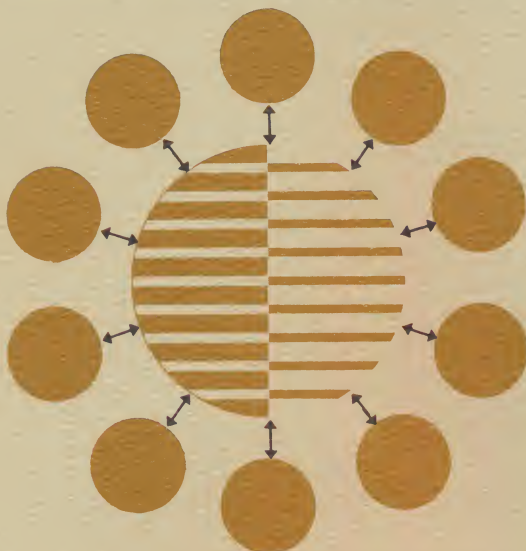
1. ONE SYSTEM — ONE USER — ONE PROBLEM
(For large, scientific problems, requiring extensive memory storage.)
2. ONE SYSTEM — MULTIPLE USERS — MULTIPLE PROBLEMS
(For time-sharing, job-shop operations, research institutes, universities.)
3. MULTIPLE SYSTEMS — MULTIPLE USERS — MULTIPLE PROBLEMS
(For large corporations, organizational "Satellite" networks, problem-mix assignments.)

The entire 6000 Series were specifically designed to work with an already-proven, overall operating system to coordinate complex data processing jobs, no matter how intricate or involved. Peripheral processors automatically eliminate "traffic jams," so that programs are processed smoothly with a reliability and speed never available before.

The results will show up in many forms . . . lower computing costs, greater information availability and, ultimately, profits. The 6000 Series' computation-per-dollar savings are unequalled in the industry. Each system within the series is a "prime mover." The information in this folder will give you an idea of what we mean.

PRIME MOVERS

From the Hardware Point of View



6000 SERIES' CIRCUITRY

CENTRAL PROCESSOR

6400—Unified Arithmetic
6600—Multi-Function Arithmetic
6800—Multi-Function Arithmetic

Each system in the **Control Data 6000 Series** is capable of handling an almost unlimited variety of peripheral equipment, and can synchronize 11 sets of instructions at once, so that it can speed through a complex problem without having to pause for each step to be completed before proceeding with the next.

Responsible for this speed, versatility and capacity is an ultra-fast central processor and memory fed by ten independent peripheral/control processors.

Primarily responsible for the speed and reliability of the processors is a new concept in circuit design . . . the silicon transistor . . . with its extended resistance to temperature extremes. Coupling the performance characteristics of this component with special high-density packaging techniques, Control Data has been able to harness high-speed electrical reactions in surprisingly compacted space, increasing not only individual data throughput, but also mass acceleration of great volumes of information with unequalled reliability.

Two types of central processors are available for the 6000 Series. The 6400 has a unified arithmetic section. The 6600 and 6800 have arithmetic sections with ten functional units:

- Increment (Two)
- Floating Add
- Fixed Add
- Shift
- Multiply (Two)
- Divide
- Boolean
- Branch

Since the operation of the central processor is isolated from all peripheral activity, it is free to perform only arithmetic functions. To further facilitate computation, all units in the multi-function sections operate in parallel so that a number of arithmetic operations can be performed concurrently. This concurrency allows the multi-function sections to accept instructions to obtain operands from the 24 internal registers (available in all three 6000 Series Systems), perform the assigned com-

putations and place results back into the internal registers . . . unhampered not only by peripheral activity, but by individual internal functions as well.

The central processor memory is a magnetic core type, arranged in independent banks of 4,096 sixty-bit words each. The 6400 gives you a choice of eight, sixteen or thirty-two banks. The 6600 is available with sixteen or thirty-two banks, and the 6800 has thirty-two banks. Memory accesses may be overlapped so that, in the absence of conflicts, they can proceed at high cycle rates. In addition, five low-order bits in the address field refer to memory bank so that successive accesses to data avoid memory conflicts, and data can be stored and accessed in numerical sequence at maximum speed.

Ten peripheral and control processors operate independently of the central processor. Each is identical and has its own memory of 4,096 twelve-bit words, can access central memory and can transfer data between central memory and its own memory. The processors are connected to the various peripheral devices through a set of data (I/O) channels, and any of the processors may select any of these channels.

Various combinations of two instruction formats are used to fit into the 6000 Series' 60-bit word. One is 15-bit and the other 30-bit in size.

The 15-bit instruction format is divided into five 3-bit fields. The first two are combined to contain the operation code. Each of the remaining 3-bit fields is used to designate an internal register address. Three bits provide sufficient addressing capability for both operands and results, since the functional units, in executing instructions, use the internal registers of the central processor, and the operation code, in every case, designates which of the internal registers is involved.

The 30-bit instruction format is divided into five sections. The first two are combined to contain an operation code. The next two are used for a result register and an operand register. The last is an 18-bit field used to hold an 18-bit constant which is used as the second operand directly, rather than as a register address.

Central Processor Speeds

6400—Major Cycle: 1000 ns
Minor Cycle: 100 ns

6600—Major Cycle: 1000 ns
Minor Cycle: 100 ns

6800—Major Cycle: 250 ns
Minor Cycle: 25 ns

PERIPHERAL/CONTROL PROCESSORS

Data Channels (I/O)

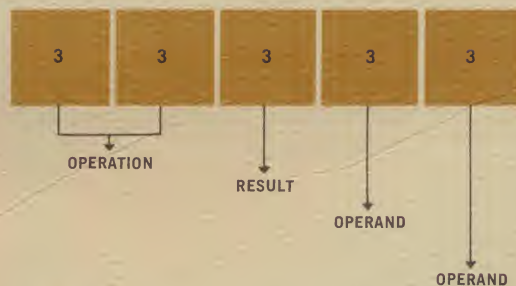
6400—12 Channels

6600—12 Channels

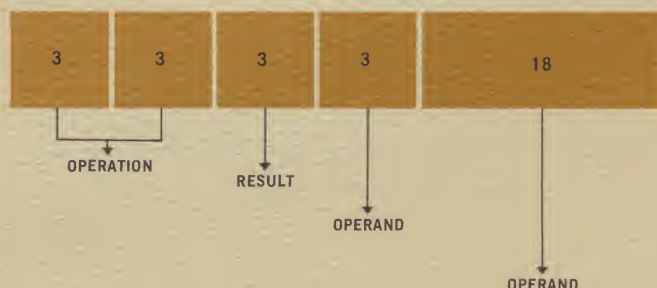
6800—12 Channels

TWO INSTRUCTION FORMATS

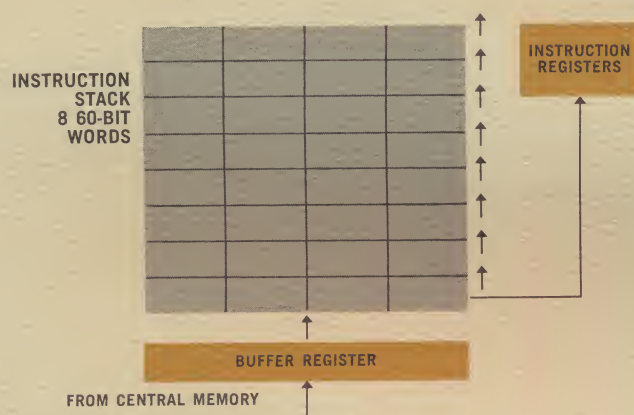
15-BIT WORD



30-BIT WORD



FLOW OF INFORMATION



FUNCTION	TYPICAL INSTRUCTION	RESULT
FETCH	$A_i = A_j + B_k$ ($i = 1-5$)	CONTENTS OF MEMORY CELL A_i PLACED IN X_i
STORE	$A_i = A_j + B_k$ $i = 6, 7$	CONTENTS OF X_i STORED IN MEMORY CELL A_i

Central Processor

The 6400 has an instruction buffer register that supplies a continuous stream of instructions into the unified arithmetic section that operates in a traditional manner.

The 6600 and 6800 have an instruction stack. Instructions flow from the central memory into the stack that contains up to eight 60-bit words of instruction contents. A record of the recent history of instructions sent to instruction registers and functional units is contained in the stack. Basically, instructions float up to the top of the stack, float off and are lost.

In the normal flow of instructions, as soon as a 60-bit word has moved from the buffer register to the bottom of the stack, the fetch of a new 60-bit word is initiated. This fetch proceeds concurrently with the flow of the previous word through the instruction registers and may be completed before the previous instruction word is completely processed.

In addition, short loops of instructions may be executed, under some circumstances, entirely within the stack, which increases system speed by reducing the probability of bank conflicts when performing concurrent memory accesses.

Here's how it works: Instructions are fetched from central memory in sequence, pass through a buffer register into the stack, flow out to the instruction registers and up through the stack. When a jump instruction is encountered in the instruction sequence, it is possible to determine if the jump destination is within the stack. If it is, the fetching of central memory instruction stops, and from that point on, instructions flow directly from the stack location into the instruction registers. This mode of operation continues until the exit condition for the loop is reached and normal fetching of instructions resumes.

Instruction Register to Functional Units From the stack in the 6600 and 6800, instructions flow into an instruction register, where it is determined which of the functional units is required to execute a particular instruction.

Central Memory to Internal Register The fetching of operands from the 6600 and 6800 central memories, and storing them in the internal registers, is performed in an interesting way. Registers "A-1" through "A-5" and "X-1" through "X-5" are linked together and are used for fetching operands. A class of instructions in the central processor repertoire performs 18-bit arithmetic in increment units. The increment units are in the set of 10 functional units and provide the capability to combine the contents of the various "A" and "B" registers to form new 18-bit numbers which can be placed in other "A" and "B" registers.

To fetch an operand from central memory requires instructing one of the increment units to perform an 18-bit arithmetic operation and to place the result in registers "A-1" through "A-5." As soon as this operation is complete, a fetch cycle is initiated and the contents of central memory at the location corresponding to the new 18-bit "A" register contents are placed in the corresponding "X" register.

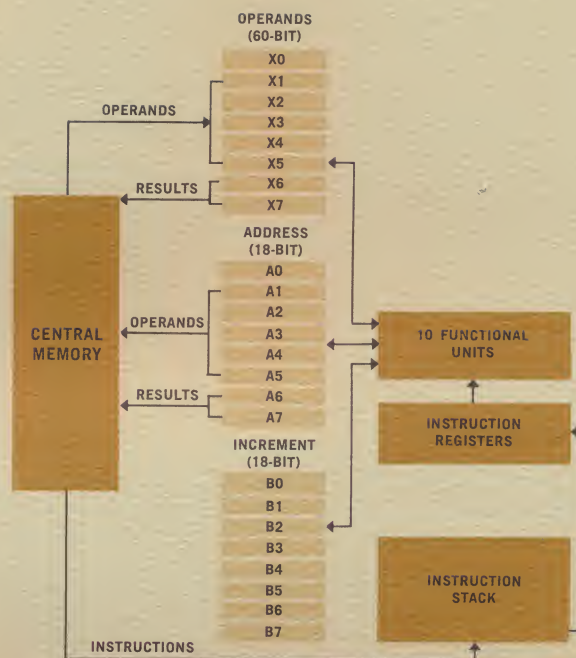
EXAMPLE: If the 18-bit arithmetic places a new quantity in the "A-2" register, the fetching cycle will place the 60-bit operand required in the "X-2" register.

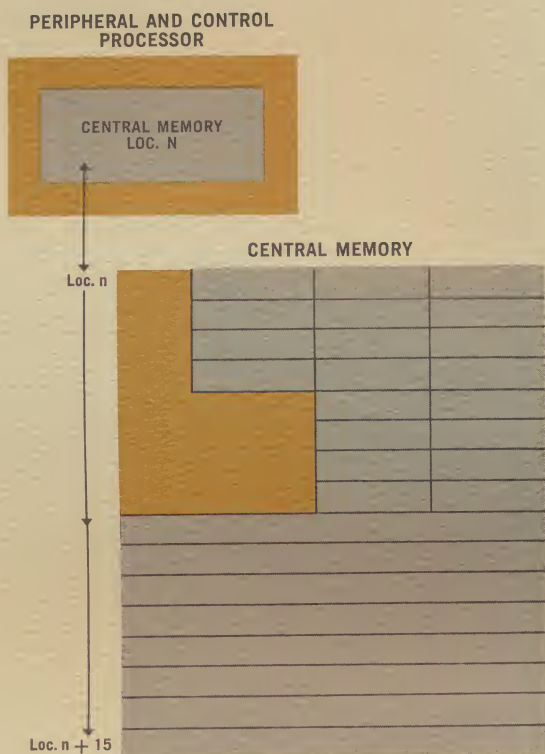
Registers "A-6, A-7, X-6 and X-7" are tied together in a corresponding way for storing results.

Peripheral/Control Processors

The revolutionary system of peripheral/control processors and data channels is identical in the entire 6000 Series, and the multiple-user system, in particular, is enhanced by the availability of these 10 independent peripheral processors. System control resides in these processors, where their private storage and ability to operate independently of the rest of the computer substantially increases system reliability. They can maintain a constant monitoring of interrupt conditions and can screen, filter, buffer and coordinate interrupt requests.

The interrupt request is executed by an exchange-jump instruction, which is in the peripheral processor repertoire.





Exchange-Jump Instruction When a peripheral processor executes an exchange jump, it must have identified a location in central memory, which is called "Location N." The location identifies the beginning of a 16-word block in central memory. In the central processor, at this time, there is a certain amount of data which can be used to describe the status of the program which is in execution in the central processor. This data includes, for example, the contents of all the internal registers, the program address and a number of other status parameters.

When the exchange jump instruction is executed, operation in the central processor is interrupted, and all of the status information is transferred from the central processor to the 16-word block identified by the number "N." At the same time, the prior contents of the 16-word block are transferred to the central memory to make up new status parameters for the start of execution of a new program.

Thus, each of the peripheral processors has the capability to interrupt a central processor program, to store in central memory all of the status information which is required to identify the status of that program, to restart it and to execute a new program in the central processor.

The relative speed of block transfer in the peripheral processors is extremely high, since each data channel can move 12 bits every microsecond.

In addition, bounds registers are set and controlled in the peripheral system control to protect relocation properties of programs for multiple users.

CONTROL DATA

6600

Three of the 16 page frames in the 6600.

PRIME MOVERS

From the Programmer-Operator Point of View

The objective in setting up a Control Data Software Package for the 6000 Series was to provide an efficiently simple, yet widely applicable overall operating system that would take full advantage of the multi-processing capabilities of the hardware itself. The system is open-ended, leaving as parameters, as many of the system "trade-offs" as possible, so that individual installations may add to and tailor the system easily to their own unique problem mixes.

Programming systems for the 6000 Series are now in the field and operating. Their ability to *tackle maximum processing assignments with minimum problems has been proven.*

The combination of central processor and multiple peripheral processors presents no problem. Programming chores are no more difficult than with traditional equipment. In many cases, they are easier. Special conveniences for the operator have been built into the system, and the use of programmer options and the regulation of a priority system offers maximum adaptability to the widely varying types of jobs that appear from installation to installation.

Put all of these elements together, add adequate memory storage capability, and you have an unbeatable combination to tackle the throughput of any system, especially when it depends upon how fast and reliably it processes a widely varying job mix on a multiple-program, multiple-user basis.

THE STANDARD SYSTEM is a single package under SIPROS (SIMultaneous PROcessing Operating System) operating control, consisting of a FORTRAN 66 Compiler, ASCENT (Assembly System for the CENTral processor), ASPER (Assembly System for the PERipheral processors) and a library of mathematical, utility and I/O routines used by all systems.

It makes little difference which of the 6000 Series' Systems you approach, the programming is absolutely the same. All advantages apply, whether you transfer programs machine to machine, upgrade total performance power by moving from 6400 to 6600 to 6800, or "gang-tackle" a massive number of complex and different problems by satellite arrangement of intermixed 6000 Series Systems.

SIPROS is an overall operating system, specifically designed to maximize multi-processing capa-

bilities. Its executive and monitoring role is contained in one of the peripheral processors, and is responsible for the control and management of all the other parts of the system, including allocation of central memory and other peripheral processors and equipment.

To aid system efficiency and ease multi-programming chores, any of the peripheral processors may be used for controlling the system as well as for buffering and processing I/O functions. Each can maintain a constant monitoring of interrupt conditions and can screen, filter, buffer and coordinate interrupt requests, as well as set and control special central memory bounds registers to protect the relocation properties of programs belonging to multiple users. In addition, they can easily be programmed to share portions of the computation load as well, although this is not intended as their normal mode of operation.

To optimize the flexibility offered by the use of ten peripheral processors, considerable attention has been paid to the use of the disk, for such functions as the storage of the library and programming systems, buffering all output to printer, etc., storing the job stack and the use of the disk as a scratch area for the programmer and system.

The use of multiple peripheral processors plus a disk aids the programmer. Normally, the programmer, using ASCENT OR FORTRAN is completely unaware of the fact that there are any number of peripheral processors performing input/output functions. He does not explicitly program them. In like manner, he need pay no special attention to the specific use of disks or tapes performing his scratch work. He simply requests scratch operations in terms of macros, defining logical records and logical tape units. Channel and peripheral equipment status is checked through the use of system macros as well.

And, although a great number of other programs will occupy a 6000 Series' System during the execution of a programmer's problem, he need not be concerned with possible interference with, or from, the other programs. SIPROS automatically takes care of questions of scheduling, loading, allocation of memory space and peripheral equipment, relocation of programs within central memory and of protection of the various parts of the program from conflicts with other programs.

SIPROS

Simultaneous PProcessing Operating System

USE OF DISK

CONTROL CARD FORMAT

The programmer is also presented with a very simple control card format. Although he can specify a large number of conditions, estimates and options for his own convenience, he is actually required to provide only a job identification card and an end-of-job card.

SIPROS also permits special programming through a mixture of central processor and peripheral processor routines, such as:

- Real-time programming, where on-line I/O must be handled.
- Additions to the operating system due to novel peripheral additions (multiple consoles, new peripherals, etc.)
- Programs where timing restrictions direct coordination between central processor execution and I/O from a peripheral device.

The programmer need only organize his deck into central processor routines and peripheral processor routines through appropriate control card. The programming system creates the necessary communication links between the two, and the SIPROS loader will automatically take care of loading the object program into the allocated central memory space and the allocated peripheral processors. The peripheral processors allocated are removed from normal SIPROS control, and "belong" to the job during execution. The "job" itself remains under SIPROS control, and the monitor watches its status and treats the entire job as any other in the system.

In this way, SIPROS is "open-ended," and a programmer using ASPER programs can actually be "adding" to the system, with no more problem than writing a trivial control or driver routine for the peripheral processor with respect to a peripheral or I/O device.

In essence, a programmer is, in effect, writing traditional programs . . . the usual programmer and system macros are provided in both systems (ASCENT or ASPER).

To ease operator assignments, communication with the system is completely handled by typewriter keyboard and two cathode ray tubes.

Status information on the flow of jobs through the system is provided along with necessary messages concerning operator action. Considerable override capability is provided, so that the operator can

SPECIAL OPERATOR CONVENIENCES

introduce or delete jobs, change their priorities, reserve equipment or interrogate the system for status information not routinely provided.

Other aids to operation include a high-level diagnostic check of hardware operation, a special method of examining selected portions of memory and the fact that the system keeps a permanent record of job history, which the operator may call at any time.

Just as the actual mix of work or job load can vary considerably from installation to installation, so too can system throughput.

Some of the parameters of the SIPROS program can be changed to adjust conveniently to individual installation requirements to gain maximum throughput economy. These parameters relate primarily to the trade-offs between time and memory space including:

- Memory Space
- I/O Buffer Size
- System Routine Space
- Disk Space Allotments
- Temporary Storage Region
- Macro Storage
- Priority Standard
- Print Output Limit
- Peripheral Processor Usage

Additionally, the programmer may select the use and regulation of a priority system provided by SIPROS. This is of particular importance in those cases where jobs are entering a system off-line, on-line or from a number of sources with varying priority requirements.

The SIPROS Parametric Priority System is divided into three classes:

1. A number of priority levels, one of which is fixed with the job, are part of its loading criteria and do not change as long as the job is in the system.
2. An equal number of priority levels are partitioned by the system internally and increased incrementally at time intervals. (The time intervals act as installation parameters.)
3. A single priority level, higher than the other two handles real-time requirements.

INSTALLATION ADAPTABILITY

FORTRAN 66... SPECIAL FEATURES

ASCENT-ASPER COMMUNICATION

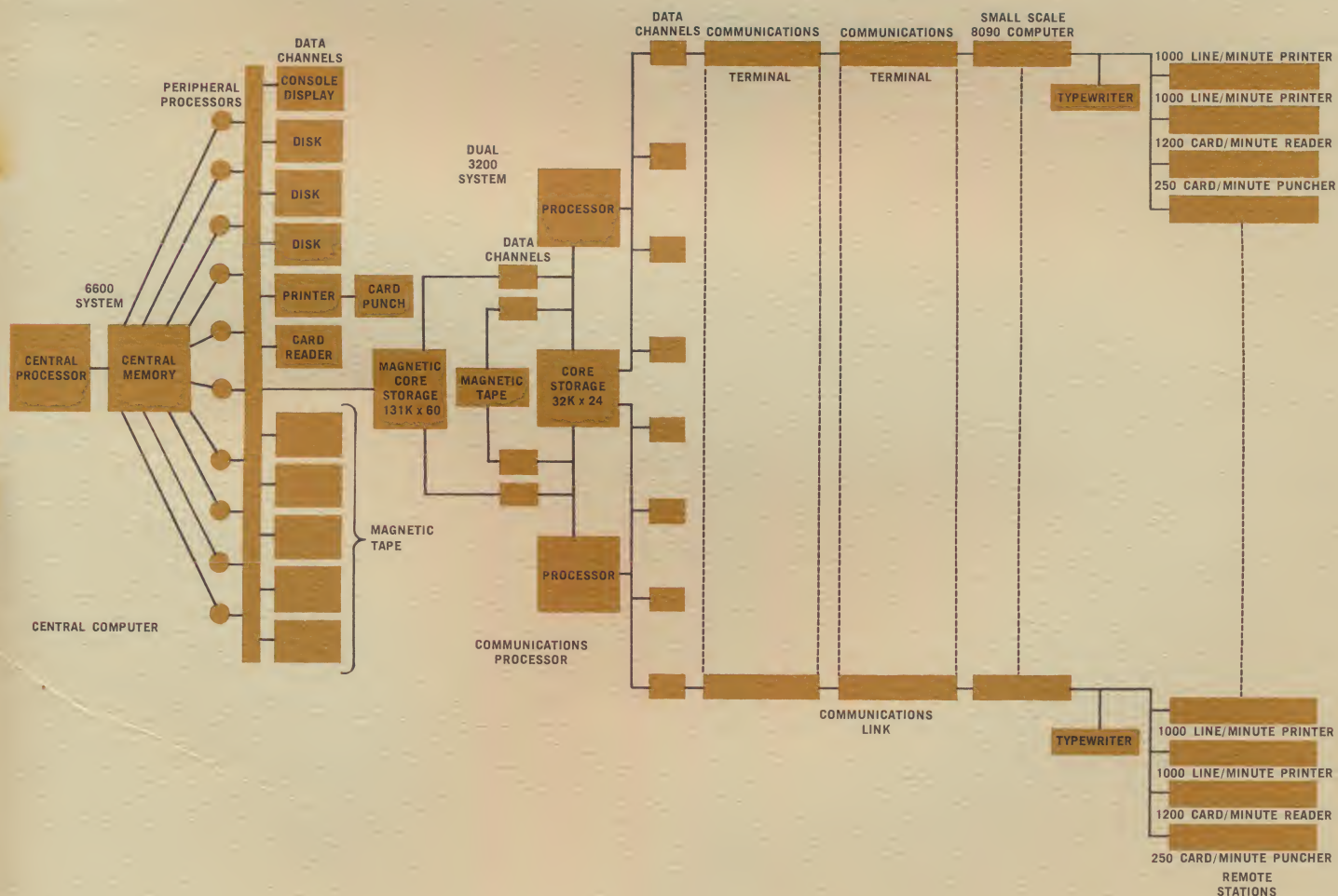
This priority system allows a user to set up a multi-program mix, where jobs are executed according to their classification. This allows any user to fit real-time jobs in with others having a high-priority requirement, holding those where time is not of the essence. In this way, system efficiency is automatically held at high levels, and takes full advantage of the speed and capability of the 6000 Series' hardware.

This FORTRAN 66 Compiler is written in the FORTRAN 63 language and is essentially upward compatible with FORTRAN IV. In addition to features already available in FORTRAN programs, two extra benefits aiming at flexibility and efficient object programs have been provided:

1. FORTRAN and ASCENT routines may be intermixed on a line-for-line basis, to reference FORTRAN names in ASCENT statements, and FORTRAN statement numbers in ASCENT. As a matter of fact, there is no distinguishable difference between coding in ASCENT with intermixed FORTRAN statements and coding in FORTRAN with intermixed ASCENT statements.
2. A code generation, optimization algorithm helps to exploit the 6000 Series' hardware design.

Coding special systems that must bring peripheral processors into explicit cooperative operation with the central processor is facilitated by the communication between ASCENT and ASPER.

During the second pass of assembly, the programming system produces relocatable binary central memory or peripheral processor programs representing the ASCENT program and the ASPER routines. The ASPER programs are separated by control cards which represent information to the loader for use at execute time. System macros, inserted by the programmer, request the assignment of a peripheral processor at execute time, and cause the system to assign a peripheral processor (removing it from the SIPROS pool), and cause the ASPER routine to be loaded. At this time, the cooperative operation between the routines in peripheral processor and central memory can proceed, and the common reference to symbols is automatically achieved by the peripheral processors' routine reference to the appropriate central memory locations during executions.



PRIME MOVERS

From the User Point of View

BIGGEST CHALLENGE

As you may have noticed by now, the biggest challenge facing a potential user of a 6000 Series' System is not one of equipment keeping up with assignments, but rather of assignments keeping up with equipment.

The unprecedented throughput speed, computational capability, extended I/O and communication facilities of the 6000 Series remove whatever time and space limitations, users have been faced with in the past. As a result, the 6000 Series invite the creative imagination to extend itself into entirely new areas of application and new solutions for old problems.

Surprisingly enough, the people involved will not be working on unfamiliar ground. Programmers and system designers can use traditional methods. There are actually less operator duties involved. And, the advanced technology of the hardware is designed for maximum reliability (as well as capability) and minimum maintenance.

THREE LEVELS OF USE

The number of possible configurations available to meet individual requirements can be considered endless. All configurations, however, may be located within three primary levels of use:

1. One System — One User — One Problem
2. One System — Multiple Users — Multiple Problems
3. Multiple Systems — Multiple Users — Multiple Problems

These three levels also characterize the growth possibilities, available in the 6000 Series, that are as endless as the configurations. The basic difference between the three systems available in the 6000 Series is one of speed only. The compatibility of programming for all three systems permits combinations and additions of computers to handle capabilities far into the future. Satellite operations are easily set up. Peripheral control processors, actually computers in their own right, may be added as satellites in units of 10, almost ad infinitum. Peripheral and communication equipment, selected from standard lists, currently available, can be used in amazing numbers, especially when memory storage is increased. Again, the growth from a basic to a more highly sophisticated system is limited only by the imagination and ingenuity of the people involved.

And, as peripheral capabilities catch up with the technology of the 6000 Series, even greater extensions can be made. A concept such as the proposed worldwide satellite communications network, presupposes worldwide processing networks, utilizing 6000 Series' Systems.

For the present, however, we are interested only in proving the processing possibilities available now. They are more than substantial. We couldn't possibly list them all here. For more information, contact your nearest Control Data office, or write for brochures available on multiple processing methods and satellite operations.

THREE PRIMARY LEVELS OF USE



ONE SYSTEM / ONE USER / ONE PROBLEM

For large, scientific problem assignments requiring extensive memory storage.



ONE SYSTEM / MULTIPLE USERS / MULTIPLE PROBLEMS

For time-sharing, job-shop operations.

For research institutes, universities, etc.



MULTIPLE SYSTEMS / MULTIPLE USERS / MULTIPLE PROBLEMS

For large corporation, government "satellite" network.

For solving problem mix chores (Real-time, off-line data processing, general data processing, communication systems, etc.).



PRIME MOVERS...

Ten Peripheral
Processors
Automatically Keep
Super-Speed Central
Processor Operating
Continuously

6400

ADVANCED
TECHNOLOGY
SETS THE PACE
FOR FUTURE
SYSTEMS

Multiple-Access
Feature Opens Door
to Mass Problem
Multi-Processing,
Plus Extensive
Remote
Hook-ups

CONTROL DATA
TOTAL
COMPUTING
SYSTEM

Traditional
Methodology
Takes Full Advantage
of World's Most
Advanced
Technology

6400 CENTRAL
PROCESSOR
Includes Unrestricted,
Computation
Unit and Your Choice
of 32k, 65k or 131k
(60-Bit Word)
Fast-Access
Storage

TRADITIONAL
PROGRAMMING
CHORES, e.g.
Control Card Format
Requires Only Job
Identification And
End-Of-Job Card

TOTAL SYSTEM:
Advanced
Hardware/Software,
Field-Proven and
Operating

UNEQUALLED
PRICE/
PERFORMANCE
RATIO

MINIMUM
SPACE
REQUIREMENT
Maximum
Operator
Convenience